Resolving security issues sometimes involves its own degree of managing people's egos

devblogs.microsoft.com/oldnewthing/20190314-00

March 14, 2019

Raymond Chen

Lots of reports come in to the Microsoft Security Response Center. Resolving them is not just a technical issue, but also a social one.

For example, somebody might report a potential vulnerability, but their proof-of-concept requires administrator privileges. Naturally, this fails the *other side of the airtight hatchway* test. But we fixed the problem anyway, not because there was any proven vulnerability, but because there wasn't proof of *lack of* a vulnerability. We studied the code and couldn't find any way to carry out the attack without administrator privileges, but were not confident in our ability to rule it out completely, so we made the fix out of an abundance of caution.

Some time later, the finder reports another potential vulnerability that also requires administrator privilege. They explained that they understood that requiring administrator privilege is normally a disqualifying factor in a vulnerability, but they noted, "You accepted my earlier vulnerability report despite it requiring administrator privilege, so I assume that you investigated the issue more closely and found a vector that didn't require administrator privilege. So here's another vulnerability report that requires administrator privilege. Maybe you can turn this into a true vulnerability, too."

This is a case of a finder creating work for us. "Here, let me report a bunch of things that are clearly not vulnerabilities as written, but I'm going to make you spend a week proving that there isn't some real vulnerability lurking beneath them that I didn't find, but for which I'm going to take credit nevertheless." Careful ego-management is required to thank the finder for their efforts, but to also politely request that they wait until they actually find something before reporting it.

Another category of managing people's egos is the case of a vulnerability report that duplicates an issue that we had already identified internally as a reliability issue, but not a security issue. A fix for the reliability issue was scheduled to go out in a week, but there was concern for the repercussions of rejecting a vulnerability report while simultaneously issuing a fix for it. The finder would observe that their rejected report was nevertheless fixed and conclude that we were silently fixing vulnerabilities without disclosing them.

There were some people who proposed reverting the reliability fix to make a clear statement that the issue was not a security issue. Presumably their idea was to hold back the reliability fix for a few months to wait for the issue to blow over, and then reintroduce it.

The security release management team decided to ship the reliability fix as scheduled, but document it as a security fix even though it isn't one. Everybody wins: Customers get a more reliable system, and the finder gets a CVE number to put on their résumé. The only loser is Microsoft: When people play "security scorecard" games and tally up the number of CVEs issued, the number in the Microsoft column is artificially inflated.

That's okay. We're used to it.

Raymond Chen

Follow